Preliminary Study of Technical Terminology for the Retrieval of Scientific Book Metadata Records

Birger Larsen
Royal School of Library and Information Science, Copenhagen, Denmark
blar@iva.dk

Christina Lioma
Computer Science, University of Copenhagen, Denmark
c.lioma@diku.dk

Ingo Frommholz
Computer Science and Technology, University of Bedfordshire U.K.
ingo.frommholz@beds.ac.uk

Hinrich Schütze
Institute for Natural Language Processing, University of Stuttgart, Germany

ABSTRACT

Books only represented by brief metadata (*book records*) are particularly hard to retrieve. One way of improving their retrieval is by extracting retrieval enhancing features from them. This work focusses on scientific (physics) book records. We ask if their technical terminology can be used as a retrieval enhancing feature. A study of 18,443 book records shows a strong correlation between their technical terminology and their likelihood of relevance. Using this finding for retrieval yields >+5% precision and recall gains.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords

Book Records, Technical Terminology

1. INTRODUCTION

Information retrieval (IR) systems often can rely on having full texts available for processing. However, there are cases when full text is not available, e.g. in commercial online bookstores or traditional libraries where material may only be available in print, or where using optical character recognition is difficult. Access to these materials is primarily realised through the supplier or the library catalogue, where documents are represented by short *book records* of metadata information, e.g. author, title, etc. The problem is that book records provide very little information, and hence they are very hard to retrieve. As a consequence, the accessibility of potentially relevant books is restricted for users. This work focuses on such books from the physics domain.

We ask whether we can increase the retrievability of physics book records by focussing on the special language in this scientific domain, as used by searchers and authors. Specifically, we model separately the technical/non-technical terminology of physics book records, motivated by the intuition that technical terminology may make a good retrieval enhancing feature. Our intuition that this modelling may benefit book record retrieval is experimentally confirmed: we find a strong correlation between the technical terminology contained in book records and their likelihood of relevance. Applying this to the retrieval of book records yields notable improvements in retrieval precision and recall.

SIGIR'12, August 12–16, 2012, Portland, Oregon, USA.
ACM 978-1-4503-1472-5/12/08.

2. TECHNICAL TERMINOLOGY AND RELEVANCE

Preprocessing. We use a collection of 18,443 physics book records with 53 queries and relevance assessments (qrels) from the iSearch dataset[1]. These book records contain basic Machine-Readable Cataloging information, e.g. title, key phrases. To identify technical terms, we part-of-speech (POS) tag the collection (including queries) with the TreeTagger[2]. We extract all terms tagged as nouns, verbs, adjectives and participles, which are the most salient POS classes, hence the most likely to be technical terms. This results in a list of 12,548 terms, which we submit to Amazon Mechanical Turk (AMT) as isolated tokens (without any context) for classification into technical/non-technical terms. Using 3 AMT users per annotation ($\geq$95% approval rate, paid approximately $0.33 per hour), 34.7% of all terms were annotated as technical, and 65.3% as non-technical, with strong inter-annotator agreement (Fleiss' $\kappa \approx 0.8$).

Technical terminology density analysis. We count the number of technical terms in each book record (referred to as document henceforth) normalised by their length - this gives the document's *terminological density*. We sort all documents by their terminological density, and we divide them into 34 bins: 33 equal-sized bins of 300 documents each, and 1 bin of the remaining 253 documents. We estimate the probability that a randomly selected relevant document belongs to a certain bin as: $p(d \in b_i|rel.) = \frac{|rel.\ d \in b_i|}{|rel.\ d|}$ and the probability that a randomly selected retrieved document belongs to a certain bin as: $p(d \in b_i|ret.) = \frac{|ret.\ d \in b_i|}{|ret.\ d|}$, where

[1] http://itlab.dbit.dk/~isearch
[2] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger

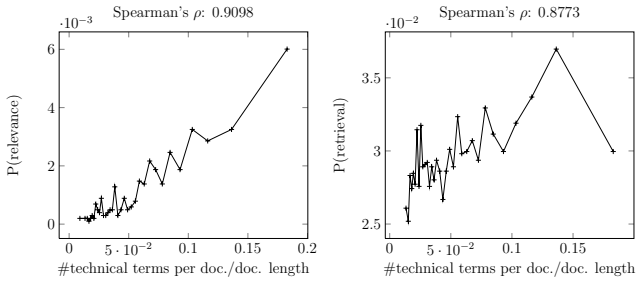

Figure 1: Normalised number of technical terms per document vs. probability of relevance (LEFT) & retrieval (RIGHT), with rank correlation coefficients.

$(rel./ret.)\, d$ is a (relevant[3]/retrieved[4]) document, b_i is the i^{th} bin, and $|\cdot|$ denotes cardinality.

Fig. 1 shows the plots of these probabilities (y axis) vs. the average terminological density in a bin (x axis). We see that $p(d \in b_i|rel.)$ varies non-randomly across bins - terminological density is positively correlated to probability of relevance ($\rho = 0.9098$). Hence, boosting the ranking of documents of higher terminological density may boost retrieval performance. The differences between how $p(d \in b_i|rel.)$ and $p(d \in b_i|ret.)$ vary across bins shows the document groups for which the retrieval model underperforms: the retrieval model has a weaker correlation to the documents' terminological density ($\rho = 0.8773$), with some instability in its performance for documents of low terminological density, and fails for documents of the highest terminological density.

Ranking model. If a user submits the query `design of biexcitonic models`, and we know that `biexcitonic` is a technical term, we hypothesise that boosting the weight of documents that contain `biexcitonic` will improve retrieval performance. We implement this boosting using Indri[5]'s combination of the Language Modeling (LM) and inference network approaches [1], which allows assigning degrees of belief to different parts of the query. This belief can be drawn from any suitable external evidence of relevance - in our case the knowledge that a search term is technical terminology. Using the *#weight* and *#combine* operators for combining beliefs, the relevance of a document d to a query q is computed as the probability that d generates q: $p(q|d) = \prod_{t \in q} p(t|d)^{\frac{w_t}{W}}$, where $W = \sum_{t \in q} w_t$, t is a term and w_t is the term's belief weight. The higher w_t is, the higher the rank of documents containing t. We apply the above equation separately for non-technical common query terms with belief weight w_{com}, and for technical query terms with belief weight w_{tec}, ($w_{com}, w_{tec} \in w_t$, $w_{com} + w_{tec} = 1$). To boost the ranking of documents containing technical terms, we increase w_{tec} at the expense of w_{com}.

Experiments. The baseline matches the documents to queries without any treatment of technical terminology using LM with Dirichlet smoothing. Our approach boosts the weight of technical terms using the same retrieval model but enhanced with belief weights as described above (TEC). We also use a pseudo-relevance feedback (FB) baseline (Indri's default FB implementation), against which we compare our approach combined with FB (TEC+FB). We measure performance with the standard TREC metrics shown in Table 1, averaged over all queries for the top 1000 results (apart from the number of relevant retrieved documents (REL.RET.) which is summed). For each metric we tune: Dir's $\mu \in \{100, 500, 800, 1000, 2000, 3000, 4000, 5000, 8000, 10000\}$; the belief weights $w_{com}, w_{tec} \in \{0.1 - 0.9\}$ in steps of 0.1 with $w_{com} + w_{tec} = 1$ at all times; FB's number of feedback documents $\in \{1, 2, 5, 10, 20\}$ and number of feedback terms $\in \{3, 5, 10, 20, 40\}$.

[3]according to qrels

[4]in top 1000 for any query by the baseline (see *Experiments*)

[5]http://www.lemurproject.org

	BPREF	NDCG	MRR	P@100	REL.RET.
BASE	28.67	23.84	28.74	02.67	185
TEC	33.97 +18.5	25.17 +5.6	29.21 +1.6	02.88 +7.9	234 +26.5
FB	36.60 +27.7	27.71 +16.2	32.46 +12.9	02.92 +9.4	222 +20.0
TEC+FB	38.24 +33.4	27.77 +16.5	32.85 +14.3	03.07 +15.0	261 +41.1

Table 1: Retrieval of our method (TEC) vs. the baseline (BASE) & pseudo-relevance feedback (FB). +% shows percent difference from the baseline.

Table 1 shows that boosting the weight of technical terminology improves retrieval at all times[6]. The biggest improvement is for REL.RET., indicating that our approach introduces to the ranking relevant documents that neither the baseline nor FB retrieve. This finding is positive, considering that our approach does not add new terms to the query - it just boosts the weight of existing query (technical) terms. Average precision benefits more when non-assessed documents are ignored in the ranking (BPREF) than when using graded relevance assessments (NDCG), possibly because NDCG gives a lower score to relevant documents that occur in the low ranks (and hence 'penalises' non-relevant documents or non-assessed documents that occur in the high ranks). Our approach benefits early precision for both the top 100 retrieved documents (P@100) and the first relevant retrieved document (MRR). We can also report that our approach outperforms the baseline and FB across the whole tuning range of μ and for $w_{tec} = 0.1 - 0.5$ (plots not included for brevity) without any outliers. The values $w_{tec} = 0.1 - 0.5$ practically correspond to applying a moderate boost to the weight of technical terminology.

3. CONCLUSION

We asked whether the retrieval of scientific books represented only by limited metadata can be improved by using their technical terminology, motivated by the empirical finding that the proportion of technical terms they contained was positively correlated with their probability of being relevant. To our knowledge this is a novel finding. We integrated this finding into the retrieval model successfully by boosting the ranking of documents containing technical terms, hence showing that our approach benefits the retrieval of book records. Future work includes using the technical term annotations to train an automatic classifier, comparing our approach against an automatic way for determining term relevance (e.g. ontologies, wikipedia pages), and testing the generality of our approach on more scientific domains.

4. REFERENCES

[1] D. Metzler and W. B. Croft. Combining the language model and inference network approaches to retrieval. *IPM*, 40(5):735–750, 2004.

[6]Results were not stat. significant when the t-test was used.